CLAIMS

1 1.   A method of transforming a class, comprising a usage
2 method accessing at least one class field, said class being
3 loadable by a class loader in an object-oriented environment,
4 said method comprising the steps of:

5   (a) creating from an original class, which comprises a
6 class field, an original-class class-initialization method,
7 and a helper class, by

8   i) converting at least one said class field to an
9   instance field and introducing the instance field into
10   said helper class; and

11   ii) converting the original-class class-initialization
12   method to a helper-class instance-initialization method
13   and introducing it into said helper class which comprises
14   a helper-class class-initialization method; and

15   (b) creating for the class a corresponding modified class
16 by converting the usage method to a modified-usage method,
17 wherein each access to the class field is replaced by an
18 invocation of an access function for fetching, for a process
19 with an instance of the helper class, from the instance, the
20 instance   field   corresponding   to   the   class   field,

21   the helper class and the modified class being loadable the
22 class loader.

1   2. The method according to Claim 1, further comprising the
2   steps of:

3   (c) creating, for the original class, a corresponding
4   modified-original class; and

5   (d) converting each original-usage method of the original
6   class to a modified original-usage method and introducing it
7   into said modified-original class, wherein each access to said
8   class field is replaced by an invocation of the access
9   function for fetching, for a process with an instance of said
10   helper class, from the instance, the instance field
11   corresponding to the class field.

1   3. The method according to Claim 2, wherein said creating
2   step (c) comprises creating, for each class field in the
3   original class, at least one of an access function, a read
4   access function and a write access function.

1   4. The method according to Claim 1, wherein the original
2   class is transformed into one helper class.

1   5. The method according to Claim 1, wherein the original
2   class is transformed into one modified class.

1   6. The method according to Claim 1, wherein transforming the
2   class is applied to a byte code.

30

1   7.   The method according to Claim 1, further comprising the
2   step of loading the helper class and the modified class by
3   use of the class loader when one of the processes is started.

1   8.   The method according to Claim 1, wherein said converting
2   substep (ii) further comprises introducing the original-class
3   class-initialization method into the modified-original class
4   and replacing the original-class class-initialization by an
5   empty method.

1   9.   The method according to Claim 1, wherein the helper-class
2   class-initialization method creates a table.

1   10.   The method according to Claim 1, further comprising the
2   step of transforming an original interface, comprising at
3   least one class field and/or an original-interface
4   class-initialization method, into a modified interface and the
5   helper class, comprising the substeps of:

6

7   (i) converting at least one said class field to an
8   instance field and introducing it into the helper
9   class; and

10   (ii)   converting   the original-interface
11   class-initialization method to a helper-class
12   instance-initialization method and introducing it
13   into the helper class which comprises a helper-class
14   class-initialization method,

15   wherein the helper class and the modified interface are
16   loadable by the class loader.

1   11. Computer readable code stored on computer readable media
2   for transforming a class in an object-oriented environment,
3   comprising:

4   a first process for creating from an original class, which
5   comprises a class field, an original-class class-
6   initialization method, and a helper class, said first process
7   comprising

8   first subprocessess for converting at least one said
9   class field to an instance field and introducing the
10   instance field into the helper class; and

11   second subprocesses for converting the original-class
12   class-initialization method to a helper-class instance-
13   initialization method and introducing it into the helper
14   class which comprises a helper-class class-initialization
15   method; and

16   a second process for creating for the class a
17   corresponding modified class by converting the usage method
18   to a modified-usage method, wherein each access to the class
19   field is replaced by an invocation of an access function for
20   fetching, for a process with an instance of the helper
21   class, from the instance, the instance field corresponding
22   to the class field,

23   the helper class and the modified class being loadable

24 the class loader.

1   12.   In a computing environment, a system for class

2 transformation, said system comprising:

3   a class comprising at least one class field, an original-

4 class class-initialization method, and a usage method

5 accessing at least one of the class fields, said class

6 residing in memory; and

7   a creator module for creating, out of said class, a helper

8 class and a modified class,

9   wherein at least one said class fields is convertable to

10 an instance field into said helper class, wherein said

11 original-class class-initialization method is convertable to

12 a helper-class instance-initialization method into said helper

13 class which comprises a helper-class class-initialization

14 method, and wherein in said usage method in said modified

15 class each access to said class field is replaceable by an

16 invocation of an access function for fetching the instance

17 field corresponding to the class field for a process with an

18 instance of said helper class, from said instance, and wherein

19 said helper class and said modified class are loadable by a

20 class loader.